Järelsõna

Loogika on järeldamise teadus: kuidas saab juba teadaolevast midagi uut järeldada ja mis on järeldamise piirid? Suures pildis taandub igasugune mõtlemine uute teadmiste ehitamisele olemasolevatest, sestap on loogika justkui teadus mõtlemise kõige üldisematest põhimõtetest, mis võiksid kehtida samamoodi nii hiirte, arvutite kui kosmiliste ülimõistuste jaoks.

Tundub, et nii vägevat eesmärki jahtides ei jaksa tegelikult kuigi kaugele jõuda. Loogika ajalugu sellist tervemõistuslikku kahtlust ka kinnitab: ajaloos on olnud kaks lühikest perioodi, kus loogika astus äkitselt pikad sammud edasi, ning kõigil muudel aegadel on toimunud aeglane areng paljude eripäraste detailide uurimise suunal.

Esimene periood - loogikateaduse teke - kestis antiik-kreekas Sokratese, Platoni ja Aristotelese eluajal. Aristotelest võibki julgeda lugeda loogika alusepanijaks, ja üle kahe tuhande aasta, kuni üheksateistkümnenda sajandi keskpaigani, ei toimunud loogikas Aristotelesega võrreldes põhimõttelist laadi edasiminekuid.

Teine periood - kaasaegse loogikateaduse sünd - algas Boole ja de Morgani töödega umbes 1850, liikus läbi Frege ja seejärel Russelli-Whiteheadi tulemuste Churchi ja Turingi arvutatavuse teooria ning Gödeli teoreemideni eelmise sajandi kolmekümnendatel aastatel. Sellal loodud kontekstis töötab loogikateadus siiani.

Arvutatavuse teooria ühelt poolt ja järeldamise ning tähenduse mudelite teooria teiselt poolt on sündinud samast eesmärgist ning elavad küll omaette elu, aga nende alused on tihedalt põimunud ja üksteist toetavad.

Arvutatavus tähendab laiemat mõistet, kui arvudega tegelemine: ta räägib meile, kuidas saab üldse mehaaniliselt järeldusi teha ja masina mõtlemistööd tegema panna. Kuidas saab programme kirjutada, arvutiga ülesandeid lahendada ja mis on mehaanilise järeldamise piirid. Kui ütleme "masin", siis ei mõtle me siin tingimata transistoridest ehitatud läptoppi. Arvutatavuse teooria jaoks on inimene, kvantarvuti ja mistahes ülimõtleja samamoodi masin, kus toimuvad küll ülikeerukad ja meile enamasti tundmatud protsessid, aga mille võimed on piiratud sellesama arvutatavuse teooria piiridega.

Gödeli teoreemid mittetäielikkusest räägivad samuti piiridest, aga teises kontekstis ja teises keeles: mis on teadmistest järeldamise piirid. Nad ei ütle meile, mis on teadmise piirid - võib olla neid ei olegi - küll aga seda, et mistahes olemasolevad teadmised ei saa meid puhta mõtlemise kaudu piiramatult kaugele viia.

Kumbki teooria ei vii iseenesest väga kaugele edasi: piiride teadasaamine ei aita masinatega kiiremini arvutada või järeldada, ning samuti ei aita Gödeli mittetäielikkuse teoreem uusi baasteadmisi hankida või mingite ülesannete jaoks sobivamaid järeldus- või lahendusmeetodeid leida. Samuti ei ütle kumbki, et me ei saa oma arvutamis- või järeldamisvõimet piiramatult tõsta: nad ütlevad eeskätt, et leidub ülesandeid, millele ei ole lõplikke lahendusi, kuitahes vägevaks meie arvuti või teadmiste hulk või järeldamisoskus ka kasvaks.

Arvutatavuse teoorias leitud piiridel ja Gödeli leitud piiridel on fundamentaalne seos, mida Gödel oma teoreemi kallal töötades veel ei teadnud, sest mõlemid valdkonnad arenesid korraga, peaaegu sünkroonis. Gödeli metoodikat mittetäielikkuse tõestamise jaoks võib vaadata kui iseäralikku,

spetsiaalse fookusega programmeerimiskeelt: Churchi ja Turingi poolt samaaegselt leiutatud universaalseid arvuti-programmeerimiskeeli ta sellal ei tundnud.

Kuidas on arvutatavuse piirid ja Gödeli mittetäielikkus seotud? Lühikokkuvõttena on arvutatavuse piiratusest suhteliselt kerge tuletada Gödeli mittetäielikkuse teoreemi, kasutamata kogu Gödeli poolt väljamõeldud keerulist mehaanikat. Sellise tuletamise aluseks on üks arvutatavuse fundamentaalseid resultaate, niinimetatud peatumisprobleem: üllatavalt lihtsalt saab tõestada, et ei ole võimalik kirjutada programmi - mistahes arvutile - mis suudaks võtta ette suvalise teise programmi, uurida seda, ja öelda meile alati ja kindlalt, kas see uuritav programm jääb lõpuks seisma või jätkab töötamist lõpmatuseni. Kuidas seda viimast väidet tõestada:

Oletame, et meil õnnestub kirjutada peatumisprobleemi lahendav programm mingis programmeerimiskeeles: *kaspeatub* olgu programm, mis võtab uurimiseks ette suvalise teise programmi (samas programmeerimiskeeles) *uuritav* ja viimase sisendandmed *sisend*. Eeldame lihtsuse mõttes, set sisendandmeteks on mingi tekst: mistahes keerulisemaid sisendandmeid, sealhulgas programme endid, saab alati esitada ühe tekstina. Programm *kaspeatub* olgu defineeritud järgmiselt:

- kui *uuritav(sisend)* peatub, siis *kaspeatub(uuritav,sisend)* vastab "jah"
- kui *uuritav(sisend)* ei peatu, siis *kaspeatub(uuritav,sisend)* vastab "ei"

Ilmselgelt oleks see program *kaspeatub*, kui teda õnnestub kirjutada, väga keeruline. Ehitame nüüd uue lihtsa programmi *vastik*, mis võtab samuti sisendiks mingi programmi *uuritav* ja kasutab programmi *kaspeatub* järgmisel viisil:

- kui *kaspeatub(uuritav,uuritav)* vastab "jah", siis *vastik(uuritav)* läheb meelega igavesse tsüklisse ehk ei peatu kunagi: seda on triviaalne programmeerida.
- kui *kaspeatub(uuritav,uuritav)* vastab "ei", siis *vastik(uuritav)* peatub koheselt. Seda on samuti triviaalne programmeerida.

Erinevalt *kaspeatub* programmist ei ole *vastik* programmil endal vaja lahendada keerulist ülesannet, vaid lihtsalt välja kutsuda *kaspeatub* programm ja teha vastavalt tema tulemusele ühte või teist.
Mis juhtub, kui anname *vastik* programmile tema enda sisendiks, ehk, käivitame *vastik(vastik)?* Siin on kaks hüpoteetilist võimalust ja kuigi me ei oska öelda, kumb neist on õige, ei ole rohkem variante kui järgmised kaks:

- *vastik(vastik)* ei peatu ehk läheb igavesse tsüklisse: siis programmi *vastik* definitsiooni järgi *kaspeatub(vastik,vastik)* vastas "jah", aga programmi *kaspeatub* definitsiooni järgi peaks *kaspeatub(vastik,vastik)* vastama "ei". Saime vastuolu, seega selline variant ei ole võimalik.
- *vastik(vastik)* peatub: siis programmi *vastik* definitsiooni järgi *kaspeatub(vastik,vastik)* vastas "ei", aga programmi *kaspeatub* definitsiooni järgi peaks *kaspeatub(vastik,vastik)* vastama "jah". Saime jällegi vastuolu, seega ka selline variant ei ole võimalik.

Kuna mõlemid hüpoteetilised variandid viisid vastuoluni, ei ole taoline olukord tegelikult võimalik, ehk, sellist programmikomplekti, nagu ülaldefineeritud *kaspeatub* ja *vastik* ei saa tegelikult olemas olla. Programmi *vastik* definitsioon on triviaalne: sellist programmi on väga lihtne kirjutada. Seega saab vastuolu tekkida ainult meie oletusest, et eelnevalt defineeritud programm *kaspeatub* - mida me ju tegelikult ei ole ehitanud - saab üldse eksisteerida. Seega ei jää üle muud, kui järeldada, et ülaldefineeritud programmi *kaspeatub* ei saa üldse olemas olla, ehk, seda ei ole põhimõtteliselt võimalik kirjutada. Teisisõnu, peatumisprobleem on mittelahenduv.

Peatumisprobleemi mittelahenduvuse tõestas esimesena Alan Turing 1936. aastal. Kurt Gödel tõestas oma mittetäielikkuse teoreemid paar aastat varem, 1931. Selgub, et peatumisprobleemi

mittelahenduvust kasutades saame anda esimesele mittetäielikkuse teoreemile palju lihtsama tõestuse, kui Kurt Gödel oma keerukat mehhaanikat kasutades:

Olgu meil loogika-aksioomide ja -reeglite kogu *F*, mis kirjeldavad mingi programmeeritava arvuti töösamme ja mille abil saame suvalise programmi *uuritav* ja tema sisendi *sisend* jaoks panna kirja teoreemi *Peatub*, mis on õige parasjagu siis, kui *uuritav(sisend)* peatub. Vaatame kahte varianti:

- Kui *Peatub* on õige, ja meie süsteem F on täielik (see tähendab, iga tegelikult õige väite jaoks on olemas tõestus) siis on põhimõtteliselt võimalik proovida järjest läbi kõikvõimalikud järelduskäigud etteantud kogust *F*, kuni jõuame teoreemi *Peatub* tõestuseni.
- Kui *Peatub* aga ei ole õige, siis peab olema õige teoreem *eitus*(*Peatub)*, ja jällegi peaksime kõigi variantide süstemaatilise järgiproovimise järel jõudma *eitus*(*Peatub)* tõestuseni.

See tähendaks aga, et suudame peatumisprobleemi lahendada, mis, nagu äsja nägime, ei ole võimalik. Seega ei saa meie etteantud loogika-aksioomide ja -reeglite kogu *F* olla täielik, teisisõnu, on olemas teoreemid, mida saame *F* keeles kirja panna ja mis on tegelikult õiged, aga mille jaoks ei saa olla süsteemis *F* tõestust. See tähelepanek ei sõltu meie valitud konkreetsetest aksioomidest ja reeglitest või arvutist/programmeerimiskeelest, ehk, ta kehtib mistahes *F* jaoks, mille abil on võimalik mistahes arvuti tegevust formaalselt kirja panna.

Järgmisena võiks küsida, kas peatumisprobleemi mittelahenduvust teades saaksime anda lihtsama tõestuse Gödeli teisele mittetäielikkuse teoreemile. Selgub, et jah, ka see on täiesti võimalik, kuigi tõestus on veidi keerulisem, kui äsjatoodu.

Nii Gödeli mittetäielikkuse teoreemid kui peatumisprobleemi mittelahenduvus kasutavad põhimõtteliselt sama nn *diagonaliseerimismeetodit*, mille tõi 1891 kasutusse Georg Cantor, tõestamaks, et erinevad lõpmatused ei ole kõik ühesuured, vaid eksisteerib lõpmatult kasvav ja üha keerukamaks hargnev lõpmatuste struktuur. Cantori tulemus oli üllatav: enne teda peeti erinevalt konstrueeritud lõpmatusi põhimõtteliselt ühesuurusteks ja ei peetud mõttekaks öelda, et üks lõpmatus võiks olla suurem, kui teine lõpmatus.

Näitena samasuurtest lõpmatustest võiks tuua positiivsete täisarvude 1, 2, 3, 4, ... lõpmatuse, negatiivsete / positiivsete täisarvude ... -4, -3, -2, -1, 0, 1, 2, 3, 4, ... lõpmatuse ja murdarvude lõpmatuse (murdarvud on kahe täisarvu jagatised). Mis üldse tähendab, et "kaks lõpmatust on samasuured"? See tähendab, et nende elementide vahel saab luua üksühese vastavuse: igale elemendile ühest lõpmatust hulgast vastab täpselt üks element teisest lõpmatust hulgast. Positiivsete / negatiivsete arvude puhul saab selle vastavuse positiivsete täisarvudega ehitada näiteks jadaga 0, 1, -1, 2, -2, 3, -3, 4, -4, ... mis sisaldab kõiki positiivseid / negatiivseid täisarve ja mille iga elemendi juurde saab kirjutada talle üksüheselt vastava positiivse täisarvu:

0, 1, -1, 2, -2, 3, -3, 4, -4, ...
1, 2, 3, 4, 5, 6, 7, 8, 9, ...

Cantor näitas juba aastal 1874, et reaalarvude puhul ei ole sellist vastavust võimalik ehitada. Mis on reaalarvud? Need on komaga arvud, millel on pärast koma kuitahes palju kohti. Reaalarvud sisaldavad muidugi nii täisarve kui murdarve, aga ka arve nagu pii ja ruutjuur kahest, mida ei saa esitada kahe täisarvu jagatisena. Tõestus ise on jällegi suhteliselt lihtne:

Oletame, et meil õnnestub ehitada kõigi reaalarvude vastavus täisarvudega. Piisab isegi, kui vaatame ainult reaalarve, mis on nulli ja ühe vahel. Vastavus tähendaks lihtsalt, et kõik sellised reaalarvud saab panna üksteise alla tabelisse, kus reanumber ongi vastav täisarv ning iga rida võib sisaldada

potentsiaalselt lõpmatult palju numbreid pärast koma. Oletame, et sellise lõpmatu tabeli saab ehitada, ehk, me saaksime seda mingil viisil lõpmatult jätkata, nii et iga reaalarv nulli ja ühe vahel sinna ka mingil hetkel jõuab. Tabel koos reanumbritega näeks välja näiteks selline, kus esimestes ridades on juhuslikult valitud näitearvud ja iga arvu ees olevat 0, (null koma) me välja ei kirjuta:

1: **5** 1 8 3 3 0 1 3 1
2: 2 **3** 0 1 9 1 0 8 3
3: 7 1 **2** 9 3 1 3 4 1
4: 9 1 2 **4** 3 2 9 0 1

Konstrueerime nüüd uue reaalarvu *X* selle tabeli diagonaalist 5 3 2 4 ... järgmisel viisil:
kui diagonaali esimene element on number A (näites 5), siis meie *X*-i esimene element on A+1 (või 0, kui A on 9), näites 6. Kui diagonaali teine element on number B (näites 3), siis meie *X*-i teine element on B+1 (või 0, kui B on 9), näites 4. Ja nii edasi: igale tabeli rea *Y* diagonaali elemendile vastab *X*-i *Y*-s element, mis on tabeli diagonaali elemendist erinev. Selliselt konstrueeritud uus arv *X* (näites 6 4 3 5 ...) on ilmselgelt samuti reaalarv nulli ja ühe vahel ja seega peaks olema ise samuti tabeli element.

Samas ei saa meie konstrueeritud *X* olla esimes reas (sest diagonaali esimese elemendi osas nad kindlasti erinevad) ega teises (sest diagonaali teise elemendi osas nad kindlasti erinevad) ega kolmas ega neljas ega samal põhjusel ka mitte ükski järgmine: igas reas on üks diagonaali element ja vastav number meie arvus *X* on meelega konstrueeritud olema sellest erinev.

Saime vastuolu: *X* kui reaalarv peaks olema tabelis, aga samas ta ei saa seal olla. Millest saab järeldada ainult üht: oletus, et kõik reaalarvud nulli ja ühe vahel saab põhimõtteliselt järjest tabelisse välja kirjutada ehk panna nad täisarvudega üksühesesse vastavusse, ei pea paika. Ilmselt ei ole reaalarvude hulk ka täisarvude hulgast väiksem, seega saab ta olla ainult suurem. Olemegi jõudnud täisarvude lõpmatusest suurema lõpmatuseni. Cantori teoreem annab järgmise sammuna veel üldisema tulemuse: mistahes hulga *H* (ka lõpmatu hulga) kõigi alamhulkade hulk on suurem, kui *H* ise. Seega on olemas lõpmatu üha suurenevate lõpmatuste ahel. See ahel sarnaneb ise omakorda täisarvude jadale, sestap on olemas harilikest arvudest üldisem termin *kardinaalarv,* mis hõlmab korraga nii harilikke täisarve kui taolist kasvavate lõpmatuste jada. Kõigele krooniks selgub, et sellise jadaga lõpmatuste struktuur ei piirdu, vaid läheb edasi üha keerulisemaks, ning on tõestatud, et erinevate lõpmatuste struktuuri ei saagi täielikult kirjeldada.

Võiks küsida, kas meie vaadatud esimese kahe - täisarvude ja reaalarvude lõpmatuste - vahel on äkki veel lõpmatusi, mis oleks täisarvude lõpmatusest suuremad ja reaalarvude lõpmatusest väiksemad? Sellel küsimusel on oma nimi - *kontiinumhüpotees* - ja siiani ei ole suudetud tõestada, kas selliseid vahepealseid lõpmatusi on või ei. Aastal 1963 tõestas Paul Cohen, et kõige tuntumast hulgateooria aksiomaatikast lühendiga *ZFC* ei ole võimalik tõestada ei kontiinumhüpoteesi õigsust ega väärust. Seega tuleks kontiinumhüpoteesi staatuse tuvastamiseks *ZFC*-le aksioome juurde lisada, samas tekib küsimus, et kust me teame, kas mingi lisatav aksioom on tegelikult õige või vale? Kõige lihtsam oleks lisada kontiinumhüpotees (või tema eitus) ise aksioomina, aga kuni me ei tea, kumb neist on õige, või kas üldse üks neist on õige, ei ole meil selliseks lisamiseks mõistlikku alust.

Olles rääkinud pikalt mittetäielikkusest, tasub pöörduda korra täielikkuse juurde. Gödeli doktoritöö aastast 1929 - kaks aastat enne mittetäielikkuse teooreemi - tõestas, näiliselt paradoksaalselt, et esimest järku predikaatarvutus on *täielik*. Mis on esimest järku predikaatarvutus? See on lausearvutuse oluline laiendus, kus on toodud sisse üldisuse ja olemasolu kvantoriga seotud muutujad ning saab rääkida asjade omavahelistest suhetest. Kaks näidet: iga X ja Y jaoks kehtib, et kui X on Y-i vend, siis Y on X-i

vend. Valemina All X All Y (vend(X,Y) <-> vend(Y,X)). Iga X, Y ja Z jaoks kehtib, et kui X on Y-i isa ja Y on Z-i isa, siis X on Z-i vanaisa. Valemina All X All Y All Z (isa(X,Y) & isa(Y,Z) -> vanaisa(X,Z)). Mis on predikaatarvutuse täielikkus? See on väide, et kui mingi predikaatarvutuse valem F on alati õige, ehk, õige mistahes struktuuri või mistahes asjaolude korral (loogikas nimetatakse mingit asjaolude seisu *mudeliks*), siis see valem F on predikaatarvutuse reeglite abil formaalselt tõestatav. Toome taolise alati õige väite näite, kasutades eelmist venna-reeglit tema ühe osana: [All X All Y (vend(X,Y) <-> vend(Y,X))] & vend(a,b) -> vend(b,a).

Predikaatarvutuse täielikkus on kasulik tulemus: saame olla kindlad, et predikaatarvutuse abil kirjapandavad väited on alati tõestatavad, kui nad on õiged. Predikaatarvutuse korrektsus - mida on palju lihtsam tõestada, kui täielikkust - ütleb lisaks, et kui valem on tõestatav, on ta ka õige. Ehk kokkuvõttes on meil võimalik predikaatarvutuse abil kirjapandavate õigete väidete korrektsust mehhaaniliselt tuvastada, otsides seni tõestust, kuni ta leiame. Paraku on siin kolm raskendavat asjaolu. Esimene on analoogiline peatumisprobleemile: ei saa olla algoritmi, mis iga mitte-alati-õige valemi korral suudaks meile lõpuks öelda, et see valem ei ole õige ja teda ei saa tõestada. Teiseks võib alati õigete valemite jaoks tõestuse otsimine olla ekstreemselt töömahukas, ning keerukamate valemite jaoks praktikas täiesti ebareaalne ka superarvutite jaoks, ka siis, kui neile anda sadu aastaid aega tõestust otsida. Kolmandaks võib keerukamate probleemide kirjapanek predikaatarvutuse keeles olla inimese jaoks ülejõukäivalt mahukas ja raske töö.

Eelpool mainitud hulgateooria aksiomaatika ZFC on väga võimas süsteem: enamik - kuid muidugi mitte kõik - teadaolevaid matemaatikatulemusi on põhimõtteliselt võimalik tuletada ZFC-st. Traditsiooniliselt pandi ZFC kirja võimsamas loogikas, kui esimest järku predikaatarvutus, kuid Gödel ehitas aastal 1940 aksiomaatika NBG, mis on ekvivalentne ZFC-ga, kuid on kirja pandud üleni esimest järku predikaatarvutuses. Ehk, enamik teadaolevat matemaatikat peaks olema tuletatav NBG-st predikaatarvutuse formaalsete tõestuste otsimise abil, mida saab põhimõtteliselt automatiseerida, kuid nagu öeldud, ei maksa loota, et realistliku aja ja reaalsete arvutite abil keerulisemaid tõestusi tegelikult leida õnnestub.

Predikaatarvutuse täielikkus räägib justkui vastu mittetäielikkuse teoreemile. Kuidas saab olla, et kui iga tegelikult õige predikaatarvutuse valemi jaoks on võimalik leida formaalne tõestus, siis korrektsete aksiomaatikate jaoks võib olla valemeid, mis on tegelikult õiged, aga mida ei saa nendest aksioomidest tõestada? Vastuolu näilikkusest arusaamise põhipunkt on tähelepanek, et kui oleme kirja pannud mingid aksioomid, siis võib eksisteerida mitmeid erinevaid asjaolude struktuure ehk mudeleid, mis neid aksioome rahuldavad. Näiteks, täisarvude tehete aksiomaatikat rahuldavad mitmed erinevad mudelid, kus täisarvud käituvad erinevalt, ometigi klappides aksioomidega. Meie aksioomid ei pruugi võimalikke maailma lihtsalt üheselt kirjeldada (ja keerukamatel juhtudel ei suudagi kirjeldada). Võime võtta aritmeetika-aksioomid ja väita seejärel, et on olemas lõpmatult suur arv, mis on suurem, kui mistahes täisarv. Ja et sellest lõpmatult suurest arvust on veelgi suurem arv. Selliseid arve me Cantori teoreemi järel just nägimegi: need on kardinaalarvud. Niisugune arvude interpretatsioon klapib aritmeetika aksioomidega sama hästi, kui interpretatsioon, kus lõpmatult suuri arve lihtsalt pole. Esimest järku predikaatarvutuses aga eri suurustega lõpmatusi ega ka lõpliku ja lõpmatu erinevust kirjeldada ei saa: selleks on vaja kõrgemat järku predikaatarvutust, kus muutujad võivad olla ka valemite tähistuseks. Viimase jaoks aga täielikkuse teoreemi enam olla ei saa. Mingis mõttes võiks öelda, et mittetäielikkuse teoreemid on võimalikud ainult seepärast, et lõpmatusi on palju.

Filosoofia jaoks on mittetäielikkuse teoreemid tõeliselt produktiivne allikas. Laias spektris järelmitest tasub aga olla ettevaatlik: üpris kerge on Gödeli teoreeme kuritarvitada mõttesuundades, kus teoreemid midagi neist näiliselt väljaloetavat tegelikult ei ütle. Teoreemid ei ütle, et on olemas tõdesid, mida ei ole

põhimõtteliselt võimalik tõestada või teada. Teoreemid ei ütle ka seda, et matemaatika on fundamentaalselt piiratud või et inimesel on potentsiaalselt suuremad võimed, kui mistahes arvutil: Gödel ise kaldus seda viimast väidet uskuma, kuid see ei tulene mittetäielikkuse teoreemidest. Sisemine tunne, et inimene on ehk võimeline jõudma mistahes tõeni, tuleb vähemalt osaliselt tundest, et meis ei ole fikseeritud aksioomikogumikke, vaid me ise muutume, ja et meil on lõpmatult aega: kui mitte ühel inimesel, siis võibolla on inimkond tervikuna pea surematu. Kas see tähendaks, et mittetäielikkus ei kehti? Kindlasti mitte. Ka hüpotees inimese otseühendusest kõiketeadjaga ja seeläbi loogikaülestest võimetest põrkub kõigepealt klassikalise küsimuse vastu, kas jumal saab teha nii suure kivi, mida ta ise tõsta ei jõua. Praktilisemas suunas vaadates tasub tähele panna, et kui lubame teha vigu, ehk vahel väita tõeseks teoreeme, mis tegelikult seda ei ole, siis saame "tõestada" mida iganes, ja sama saaks teha ka primitiivne masin, mis genereerib "tõdesid" lihtsalt täringuviske abil.

Küll aga ütlevad Gödeli teoreemid, et ei saa olla fikseeritud hulka tõdesid, millest kõik teised tulenevad, ja ehk ütlevad nad ka seda, et me ei saa isegi oma matemaatilistes veendumustes alati päris kindlad olla: ei saa olla kindlust, et aksioomid või teadmised, millest neid veendumusi järeldame, on tingimata õiged. Matemaatika on mingis mõttes väga lahtine: tõsikindlat alust, millele toetuda, tegelikult ei ole. Mittetäielikkus heidab seeläbi valgust klassikalisele eristusele analüütiliste lausete (ehk definitsiooni järgi õigete lausete, nagu "poissmehed ei ole abielus" või "2+2=4") ja sünteetiliste lausete (tegelikku asjade seisu kirjeldavad laused nagu "praegu on aasta 2020" või "maakera on ümmargune") vahel. Quine skepsis sellise vahetegemise mõttekusele toetub vähemalt osaliselt arusaamale, et keerukad analüütilised väited ei ole intuitiivselt selged ja ei pruugi olemas olla tõsikindlat baasi, millele toetudes neid väita. Kas väide "kontiinumhüpotees on õige" on analüütiline, on tõsiselt kaheldav.